Certain of claims 1-30 were rejected for alleged lack of statutory subject matter, alleged lack of utility, alleged indefiniteness, anticipation and/or obviousness. To the extent that the rejections are applied to the amended claims, Applicants traverse, as noted in detail below.

ELECTION/RESTRICTION.

Pursuant to a restriction requirement made final, Applicants cancel claims 31-44 with entry of this amendment. Please note, however, that Applicants reserve the right to file subsequent applications claiming the canceled subject matter and the claim cancellations should not be construed as abandonment or agreement with the Examiner's position in the Office Action.

INFORMATION DISCLOSURE STATEMENTS.

Applicants thank the Examiner for the indication that the IDS of December 7, 2001 was considered. The Examiner also indicated that the references sent with the IDS of July 2, 2001 did not correspond to the IDS; accordingly, these references will be hand delivered to the Examiner under separate cover to ensure that they reach the Examiner and can be properly considered. Similarly, it was indicated that the references sent with the IDS sent August 11, 2001 were lost by the office. These references will also be separately hand delivered to the Examiner.

AS AMENDED, THE CLAIMS ARE DEFINITE

Claims 5, 10, 14, 22 and 30 were rejected under 35 U.S.C. §112, second paragraph, as allegedly indefinite for various minor errors in antecedence and phraseology. These errors have been corrected, essentially as helpfully suggested by the Examiner. Accordingly, the rejections should be withdrawn.

THE CLAIMS MEET THE REQUIREMENTS OF 35 USC § 101

Claims 17-30 were rejected under 35 USC § 101 for allegedly not being directed to statutory subject matter. The claims have been amended, as helpfully suggested by the Examiner, to overcome this rejection. Accordingly, the rejection should be withdrawn.

Claims 1-30 were also rejected under 35 U.S.C. §101 for alleged lack of utility. To the extent that the rejection is maintained in light of the amendments, e.g., to claim 1, Applicants respectfully traverse.

As an initial matter, the Examiner is respectfully directed to the specification at page 39, line 24- page 43, line 28, where at least 5 different classes of specific and credible utilities for the

character strings and label generation systems of the invention are set forth in detail. Any of the specific and credible utilities detailed in the specification and specifically set forth meet the utility requirements of 35 USC § 101.

For example, the sequence strings of the invention are specifically useful as tags for indexing information. The sequences are particularly well suited to this task, as a result of their ability to differentiate the relatedness of the information to be tagged (e.g., by their degree of relatedness to one another). This enables one to use similarity tools to group items labeled by the tags. This utility is specifically set forth in the specification (e.g., page 40, line 30- page 41, line 12), and would clearly work for the asserted utility. Moreover, this is not a "throw away" utility, as the tags generated by the method provide a specific advantage over tags in the prior art, by providing the ability to establish relational information—and it will immediately be appreciated by one of skill that the ability to index related items, e.g., in a relational database, is of immediate practical utility.

Other utilities that are specifically set forth include the use of sequence strings generated according to the methods herein as reference objects in database searches (page 41, lines 13-24); use of the strings to identify structural motifs conferring specific molecular properties (page 41, line 25- page 42, line 18), and use of the strings in genetic algorithms, and the use of the strings to drive synthetic machinery (page 43, lines 19-29).

With respect to each of these asserted utilities, the Examiner is respectfully reminded that "A specification that contains disclosure of utility that corresponds in scope to the subject matter sought to be patented must be taken as sufficient to satisfy the utility requirements 35 USC § 101 for the entire claimed subject matter unless there is reason for one skilled in the art to question the objective truth of the statement of utility or its scope." Ex parte Rubin 5 USPQ2d 1461 (BPAI 1989). Nothing in the rejection sets forth any reasons to doubt the asserted utilities that are provided.

For example, with respect to the last enumerated specific utility, libraries of related molecules are widely understood to be extremely valuable commercial products with well recognized specific utilities. That is, such libraries have long been recognized to themselves be commercially valuable reagents. The assertion that such libraries are useful only for conducting further research misinterprets the meaning of Brenner, which was that an invention must provide a benefit available in currently available form (i.e., without further research), not that research reagents are somehow not entitled to patent protection. *See*, e.g., Brenner v. Manson 148 USPQ

689,695 (US Supreme Ct. 1966). Clearly they are—thousands of valuable patents to research reagents and research tools have issued since the decision in Brenner. The Courts have listed use of compositions for use by researchers as meeting the requirements for utility. In Re Hartop and Brandes 135 USPQ 419, 425 (CCPA 1962). Indeed, it seems immediately apparent that *anything* that includes a credible asserted potentially commercial feature must meet the utility requirements, given that even things that are not commercially salable can meet it. *See*, In Re Langer 183 USPQ 288, 298 (CCPA 1974). Applicants have asserted several utilities that are in currently available form, explaining how various products that fall under the claims are commercially valuable, requiring no further research for the benefits of the invention to be realized. The fact that some of these utilities are themselves uses as reagents for research is simply not relevant to the question of whether the utilities are adequate to meet the requirements of 35 USC § 101.

Because the invention has several asserted, credible, substantial, and specific utilities, the rejection should be withdrawn.

SUN DOES NOT ANTICIPATE THE CLAIMS

Claims 1-5, 12-13, 14-17, 24-25 and 26-27 were rejected under 35 U.S.C. §102(a) for alleged anticipation by Sun (Journal of Computational Biology 6(1):77-90). To the extent that the rejections are applied to the amended claims, Applicants respectfully traverse.

Sun does not anticipate the claims as amended. Sun proposes a mathematical model of DNA shuffling and proposes the application of the model to optimize certain aspects of DNA shuffling experiments and, e.g., for physical mapping. This model and the proposed applications simply do not provide the limitations of the claims.

For example, amended claim 1 requires "making one or more product biological molecule corresponding to one or more of the product strings." Sun performs no such operation. Instead, Sun allegedly provides a mathematical model for DNA shuffling—there is no connection between the generation of product strings and the synthesis of actual molecules. At most, Sun suggests that one might bias a shuffling reaction by putting different concentrations of DNA molecules into a shuffling reaction, based upon an unspecified prediction of the model (Sun, section 5.2). Even if Sun had actually performed this step (which was not the case), this is not equivalent to making a particular product DNA as required by the claims, nor does it suggest any way of doing so.

In fact, Applicants are unable to find any passage in Sun which relates directly to the claimed methods. Amended claim 1 requires:

1. A method of producing molecules represented by concatenated strings, said method comprising:

i) encoding two or more related biological molecules into a data structure of initial character strings to provide a collection of two or more different initial character strings wherein each of said biological molecules comprises at least about 10 subunits;

ii) selecting at least two substrings from said initial character strings;

iii) concatenating said substrings to form one or more product strings about the same length as one or more of the initial character strings;

iv) adding the product strings to a data structure to populate a data structure of product [collection of] strings;

v) optionally repeating steps (i) or (ii) through (iv) using one or more of said product strings as an initial string in the collection of initial character strings and;

vi) making one or more product biological molecule corresponding to one or more of the product strings.

Nothing in Sun relates, e.g., to selecting substrings, concatenating substrings to make product strings, adding the product substrings to a collection of strings, repeating the steps, or, as discussed above, making the product substrings, etc.. Instead, Sun teaches algorithms that rely, in an abstract sense, upon distances between mutations to be recombined, the length of overlap needed between fragments to be recombined and other physical parameters. The actual manipulation of strings of sequences simply is not described in the reference, nor is it, in any way, a feature of the model. Sun does not select substrings of sequences to be recombined, nor does the reference teach concatenation of sequences or addition of product strings to a collections of such strings, nor making biological molecules corresponding to the product strings.

The passages cited by the Examiner (e.g., pp. 79-80 and pp. 86) do not provide any teachings that seem particularly related to the claimed invention. For example, Figure 1 on page 79 is a background figure describing physical shuffling generally (this figure is referenced in the introduction section of the paper at page 78, paragraph 4, rather than with respect to the description of the model). The text for pages 79-80 provide a mathematical model of shuffling based on the number of fragments (N) to be shuffled, the coverage to be obtained (c), the fragment length that must be shared for annealing to occur (theta), the number of expected regions, the number of regions to be reassembled, etc. Applicants are unable to find any description in the reference specifically directed to the claimed invention, which relates, e.g., to a particular way of manipulating sequence strings and systems for doing so. Accordingly, the rejection should be withdrawn.

THE CLAIMS ARE NOT OBVIOUS

Claims 1-30 were rejected under 35 U.S.C. §103(a) as allegedly obvious in light of Sun. Applicants respectfully traverse the rejections for the reasons noted above (e.g., Sun does not teach the limitations of the claims) and below.

Applicants note that the rejection does not (and cannot) set forth a *prima facie* case of obviousness. The reasons for this are straightforward—as noted above, Sun simply does not teach the actual limitations of the claims. Instead, the reference provides a mathematical model that can, allegedly, be used to model aspects of DNA shuffling. The model itself does not perform any of the steps of Applicants' claimed methods or provide any relevant computer program product code for doing so, and, thus, no real relationship between Sun and the claimed invention can be made out.

Put another way, the model proposes complex mathematical relationships that can be used to predict, e.g., whether two sequences are likely to recombine in a shuffling reaction and the fraction of reassembled molecules that are likely to have a given mutation configuration (*see*, e.g., page 86, section 5.1). This ability to predict whether a mutation configuration is likely to occur is not the result of a particular way of manipulating sequences, e.g., as according to the claimed invention—instead, it is the result of a set of assumptions about components of a shuffling reaction that are embodied in mathematical formulae (*See*, e.g., Sun, Fig. 2). No character string manipulation needs to be performed to use the model, nor is any such manipulation taught. No computer program product of any kind is taught by Sun, let alone one that includes the code that is required by claim 17.

Because the reference simply does not teach the limitations of the claims, the rejection should be withdrawn.

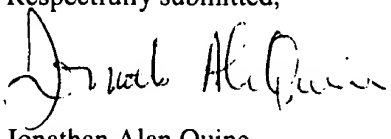
CONCLUSION

In view of the foregoing, Applicants believes all claims now pending in this application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

In the event that any issues of substance are perceived to remain, Applicants request an Examiner Interview *prior* to submission of any additional Action by the Office. If a telephone conference would expedite prosecution of this application, the Examiner is invited to telephone the undersigned at (510) 337-7871.

LAW OFFICES OF JONATHAN ALAN QUINE
P.O. BOX 458
Alameda, CA 94501
Tel: 510 337-7871
Fax: 510 337-7877

Respectfully submitted,



Jonathan Alan Quine
Reg. No: 41,261

APPENDIX A

"MARKED UP" CLAIMS ILLUSTRATING THE AMENDMENTS MADE TO THE CLAIMS OF 09/495,668 WITH ENTRY OF THIS AMENDMENT

1 (Amended). A method of **producing molecules represented by concatenated strings** **[populating a data structure with a plurality of character strings]**, said method comprising:

i) encoding two or more **related** biological molecules into **a data structure of initial** character strings to provide a collection of two or more different initial character strings wherein each of said biological molecules comprises at least about 10 subunits;

ii) selecting at least two substrings from said **initial** character strings;

iii) concatenating said substrings to form one or more product strings about the same length as one or more of the initial character strings;

iv) adding the product strings to a **data structure to populate a data structure of product** **[collection of]** strings; **[and]**

v) optionally repeating steps (i) or (ii) through (iv) using one or more of said product strings as an initial string in the collection of initial character strings**, and;**

vi) making one or more product biological molecule corresponding to one or more of the product strings.

2 (Amended). The method of claim 1, wherein said encoding comprises encoding **[one]** **two** or more nucleic acid sequences into said character strings.

3 (Amended). The method of claim 2, wherein said **[one]** **two** or more nucleic acid sequences comprise a nucleic acid sequence encoding a **[known]** **naturally occurring** protein.

4 (Amended). The method of claim 1, wherein said encoding comprises encoding **[one]** **two** or more amino acid sequences into said character strings.

5 (Amended). The method of claim 4, wherein said one or more amino acid sequences comprise **[a nucleic acid]** **an amino acid** sequence encoding a **[known]** **naturally occurring** protein.

10 (Amended). The method of claim 1, wherein said selecting comprises aligning two or more of said initial character strings to maximize pairwise identity between two or more substrings of the **initial** character strings, and selecting a character that is a member of an aligned pair for the end of one **of the two or more** substrings.

11 (Amended). The method of claim 1, wherein said product strings are added to the **[collection]** **data structure of product strings** only if they have greater than 30% sequence identity with the initial strings.

12 (Amended). The method of claim 1, wherein said method further comprises randomly altering one or more characters of said **initial or product** character strings.

14 (Amended). The method of claim 1, wherein said **en**coding, selecting, or concatenating is performed on an internet site.

15 (Amended). The method of claim 1, wherein said **en**coding, selecting, or concatenating is performed on a server.

16 (Amended). The method of claim 1, wherein said **en**coding, selecting, or concatenating is performed on a client linked to a network.

17 (Amended). A computer program product **on a computer readable media** comprising computer code that**:**

i) encodes two or more [a] biological molecules into **initial** character strings to provide a collection of two or more different initial character strings wherein each of said biological molecules comprises at least about ten subunits;

ii) selects at least two **initial** substrings from said character strings;

iii) concatenates said substrings to form one or more product strings about the same length as one or more of the initial character strings;

iv) adds the product strings to a **[collection of strings] data structure to populate a data structure of product strings**; **[and]**

v) optionally repeats steps (i) or (ii) through (iv) using one or more of said product strings as an initial string in the collection of initial character strings**; and,**

vi) directs the production of one or more product biological molecule corresponding to the product strings.

18 (Amended). The **computer** program **product** of claim 17, wherein said two or more biological molecules are nucleic acid sequences.

19 (Amended). The **computer** program **product** of claim 17, wherein said two or more biological molecules are nucleic acid sequences of **[known] naturally occurring** proteins.

20 (Amended). The **computer** program **product** of claim 17, wherein said two or more biological molecules are amino acid sequences.

21 (Amended). The **computer** program **product** of claim 17, wherein said biological molecules have at least 30% sequence identity.

22 (Amended). The **computer** program **product** of claim 17, wherein said **computer** code selects substrings such that the ends of said substrings occur in string regions of about three to about twenty characters that have higher sequence identity with **[the] a** corresponding region of another of said initial character strings than the overall sequence identity between the **[same]** two **sub**strings.

23 (Amended). The **computer** program **product** of claim 17, wherein said **computer** code selects substrings such that the ends of said substrings occur in predefined motifs of about 4 to about 8 characters.

24 (Amended). The **computer** program **product** of claim 17, wherein said **computer** code selects and concatenates substrings from two different initial strings such that the concatenation occurs in a region of about three to about twenty characters having higher sequence identity between said two different initial strings than the overall sequence identity between said two different initial strings.

25 (Amended). The **computer** program **product** of claim 17, wherein **the computer** code selects substrings by aligning two or more of said initial character strings to maximize pairwise identity between two or more substrings of the character strings, and selecting a character that is a member of an aligned pair for the end of one substring.

26 (Amended). The **computer** program **product** of claim 17, wherein said product strings are added to the collection only if they have greater than 30% identity with the initial strings.

27 (Amended). The **computer** program **product** of claim 17, wherein said **computer** code additionally randomly alters one or more characters of said character strings.

28 (Amended). The **computer** program **product** of claim 27, wherein said **computer** code additionally randomly selects and alters one or more occurrences of a particular preselected character in said character strings.

29 (Amended). The **computer** program **product of** claim 17, wherein said **computer** code is stored on media selected from the group consisting of magnetic media, optical media, optomagnetic media.

30 (Amended). The **computer** program **product of** claim 17, wherein said **computer** code is in dynamic or static memory of a computer.

45 (New). The method of claim 1, wherein the initial character strings of (i) are related.

46 (New). The method of claim 1, further comprising physically screening the molecule(s) represented by the product strings for one or more desired properties.

47 (New). The method of claim 1, further comprising determining a computationally predicted property for molecules represented by the product strings.

48 (New). The method of claim 1, wherein the molecules represented by the product strings are made in parallel in an array of vessels.

49 (New). The method of claim 1, wherein the molecules represented by the product strings are made by assembly of oligonucleotides.

50 (New). The method of claim 1, further comprising testing members of the data structure of product strings for a particular property and determining an optimal combination of sequences using multi-variate analysis.

51 (New). The computer program product of claim 17, wherein the initial character strings of (i) are related.

52 (New). The computer program product of claim 17, wherein the code instructs physical screening of the molecule(s) represented by the product strings for one or more desired properties.

53 (New). The computer program product of claim 17, wherein the code instructs determination of a computationally predicted property for molecules represented by the product strings.

54 (New). The computer program product of claim 17, wherein the molecules represented by the product strings are made in parallel in an array of vessels.

55 (New). The computer program product of claim 17, wherein the molecules represented by the product strings are made by assembly of oligonucleotides.

56 (New). The computer program product of claim 17, wherein the code tests members of the data structure of product strings for a particular property and determines an optimal combination of initial sequences using multi-variate analysis.

APPENDIX B

CLAIMS PENDING IN USSN 09/495,668 WITH ENTRY OF THIS AMENDMENT

1 (Amended). A method of producing molecules represented by concatenated strings, said method comprising:

i) encoding two or more related biological molecules into a data structure of initial character strings to provide a collection of two or more different initial character strings wherein each of said biological molecules comprises at least about 10 subunits;

ii) selecting at least two substrings from said initial character strings;

iii) concatenating said substrings to form one or more product strings about the same length as one or more of the initial character strings;

iv) adding the product strings to a data structure to populate a data structure of product strings;

v) optionally repeating steps (i) or (ii) through (iv) using one or more of said product strings as an initial string in the collection of initial character strings, and;

vi) making one or more product biological molecule corresponding to one or more of the product strings.

2 (Amended). The method of claim 1, wherein said encoding comprises encoding two or more nucleic acid sequences into said character strings.

3 (Amended). The method of claim 2, wherein said two or more nucleic acid sequences comprise a nucleic acid sequence encoding a naturally occurring protein.

4 (Amended). The method of claim 1, wherein said encoding comprises encoding two or more amino acid sequences into said character strings.

5 (Amended). The method of claim 4, wherein said one or more amino acid sequences comprise an amino acid sequence encoding a naturally occurring protein.

6. The method of claim 1, wherein said biological molecules have at least 30% sequence identity with each other.

7. The method of claim 1, wherein said selecting comprises selecting substrings such that the ends of said substrings occur in string regions of about 3 to about 20 characters that have higher sequence identity with the corresponding region of another of said initial character strings than the overall sequence identity between the same two strings.

8. The method of claim 1, wherein said selecting comprises selecting substrings such that the ends of said substrings occur in predefined motifs of about 4 to about 8 characters.

9. The method of claim 1, wherein said selecting and concatenating comprises concatenating substrings from two different initial strings such that the concatenation occurs in a region of about three to about twenty characters having higher sequence identity between said two different initial strings than the overall sequence identity between said two different initial strings.

10 (Amended). The method of claim 1, wherein said selecting comprises aligning two or more of said initial character strings to maximize pairwise identity between two or more substrings of the initial character strings, and selecting a character that is a member of an aligned pair for the end of one of the two or more substrings.

11 (Amended). The method of claim 1, wherein said product strings are added to the data structure of product strings only if they have greater than 30% sequence identity with the initial strings.

12 (Amended). The method of claim 1, wherein said method further comprises randomly altering one or more characters of said initial or product character strings.

13. The method of claim 12, wherein said method further comprises randomly selecting and altering one or more occurrences of a particular preselected character in said character strings.

14 (Amended). The method of claim 1, wherein said encoding, selecting, or concatenating is performed on an internet site.

15 (Amended). The method of claim 1, wherein said encoding, selecting, or concatenating is performed on a server.

16 (Amended). The method of claim 1, wherein said encoding, selecting, or concatenating is performed on a client linked to a network.

17 (Amended). A computer program product on a computer readable media comprising computer code that:

i) encodes two or more biological molecules into initial character strings to provide a collection of two or more different initial character strings wherein each of said biological molecules comprises at least about ten subunits;

ii) selects at least two initial substrings from said character strings;

iii) concatenates said substrings to form one or more product strings about the same length as one or more of the initial character strings;

iv) adds the product strings to a data structure to populate a data structure of product strings;

v) optionally repeats steps (i) or (ii) through (iv) using one or more of said product strings as an initial string in the collection of initial character strings; and,

vi) directs the production of one or more product biological molecule corresponding to the product strings.

18 (Amended). The computer program product of claim 17, wherein said two or more biological molecules are nucleic acid sequences.

19 (Amended). The computer program product of claim 17, wherein said two or more biological molecules are nucleic acid sequences of naturally occurring proteins.

20 (Amended). The computer program product of claim 17, wherein said two or more biological molecules are amino acid sequences.

21 (Amended). The computer program product of claim 17, wherein said biological molecules have at least 30% sequence identity.

22 (Amended). The computer program product of claim 17, wherein said computer code selects substrings such that the ends of said substrings occur in string regions of about three to about twenty characters that have higher sequence identity with a corresponding region of another of said initial character strings than the overall sequence identity between the two substrings.

23 (Amended). The computer program product of claim 17, wherein said computer code selects substrings such that the ends of said substrings occur in predefined motifs of about 4 to about 8 characters.

24 (Amended). The computer program product of claim 17, wherein said computer code selects and concatenates substrings from two different initial strings such that the concatenation occurs in a region of about three to about twenty characters having higher sequence identity between said two different initial strings than the overall sequence identity between said two different initial strings.

25 (Amended). The computer program product of claim 17, wherein the computer code selects substrings by aligning two or more of said initial character strings to maximize pairwise identity between two or more substrings of the character strings, and selecting a character that is a member of an aligned pair for the end of one substring.

26 (Amended). The computer program product of claim 17, wherein said product strings are added to the collection only if they have greater than 30% identity with the initial strings.

27 (Amended). The computer program product of claim 17, wherein said computer code additionally randomly alters one or more characters of said character strings.

28 (Amended). The computer program product of claim 27, wherein said computer code additionally randomly selects and alters one or more occurrences of a particular preselected character in said character strings.

29 (Amended). The computer program product of claim 17, wherein said computer code is stored on media selected from the group consisting of magnetic media, optical media, optomagnetic media.

30 (Amended). The computer program product of claim 17, wherein said computer code is in dynamic or static memory of a computer.

45 (New). The method of claim 1, wherein the initial character strings of (i) are related.

46 (New). The method of claim 1, further comprising physically screening the molecule(s) represented by the product strings for one or more desired properties.

47 (New). The method of claim 1, further comprising determining a computationally predicted property for molecules represented by the product strings.

48 (New). The method of claim 1, wherein the molecules represented by the product strings are made in parallel in an array of vessels.

49 (New). The method of claim 1, wherein the molecules represented by the product strings are made by assembly of oligonucleotides.

50 (New). The method of claim 1, further comprising testing members of the data structure of product strings for a particular property and determining an optimal combination of sequences using multi-variate analysis.

51 (New). The computer program product of claim 17, wherein the initial character strings of (i) are related.

52 (New). The computer program product of claim 17, wherein the code instructs physical screening of the molecule(s) represented by the product strings for one or more desired properties.

53 (New). The computer program product of claim 17, wherein the code instructs determination of a computationally predicted property for molecules represented by the product strings.

54 (New). The computer program product of claim 17, wherein the molecules represented by the product strings are made in parallel in an array of vessels.

55 (New). The computer program product of claim 17, wherein the molecules represented by the product strings are made by assembly of oligonucleotides.

56 (New). The computer program product of claim 17, wherein the code tests members of the data structure of product strings for a particular property and determines an optimal combination of initial sequences using multi-variate analysis.